Amendments to the Specification

Please replace paragraph beginning at **page 2, line 9**, with the following amended paragraph.

FIG. 1 is a block diagram illustrating a system using the conventional DMA. In FIG. 1, an advanced micro-controller bus architecture (AMBA) produced by Advanced RISC Machines ~~(ARM) Ltd.~~(ARM) Ltd. is shown. The AMBA includes an advanced high-performance bus ~~(AHB) operating~~(AHB) operating at a high frequency and an advanced peripheral bus (APB) operating at a low frequency. An AHB-APB bridge 108 is coupled between the AHB and the APB so that a high-speed bus and a low-speed bus can exchange data. Referring to the remaining system elements shown in FIG. 1, an AHB block includes a central processing unit (CPU) 100, a first storage unit 102, a DMA device 104 and a bus arbiter 106. The APB block includes a second storage unit 110 and an input/output (I/O) ~~(-)~~ unit 112. The I/O unit 112 includes a universal serial bus ~~(USB),~~(USB), a keypad, a universal asynchronous receiver/transmitter ~~(UART),~~(UART), among other items. In order for data stored in the first storage unit 102 to be transferred to the second storage unit 110, all operations are controlled and processed by the CPU 100. When the data stored in the first storage unit 102 is transferred to the second storage unit 110, the CPU 100 reads the data stored in the first storage unit 102. The CPU 100 performs a necessary operation before the read data is transferred to the second storage unit 110. However, when a large amount of data is transferred from the first storage unit 102 to the second storage unit 110, it is difficult for the CPU 100 to appropriately perform a set of operations. There is a problem in that the CPU 100 must perform many tasks associated with the system other than data transmission operations. To address this problem, the DMA device 104 has been developed.

Please replace paragraph beginning at **page 3, line 9**, with the following amended paragraph.

FIG. 2 is a block diagram illustrating the internal structure of the DMA device 104 shown in FIG. 1. As shown in FIG. 2, the DMA device 104 includes a control register, ~~an~~a source address register ~~(SAR))~~(SAR), a destination address register ~~(DAR),~~(DAR), a transfer count register (TCR) ~~(-)~~, a first-in-first-out (FIFO) buffer, a bus controller, an interface, among other

items. The SAR is a register for designating an initial source address where data is read from the first storage unit 102. The DAR is a register for designating an initial destination address where the DMA device 104 first writes the data read from the first storage unit 102 to the second storage unit 110. The TCR is a register for designating the number of writing operations when the DMA device 104 writes the data read from the first storage unit 102 to the second storage unit 110. Further, the control register performs a control operation according to a determination as to whether an address value associated with a reading operation must be incremented, decremented or fixed when data is read in a source address of the first storage device 102 and then the next data is read. Furthermore, the control register performs a control operation according to a determination as to whether an address value associated with a writing operation must be incremented, decremented or fixed when data is written in a destination address of the first storage device 102 and then the next data is written. When data is transferred from the first storage unit 102 to the second storage unit 110, the control register performs a control operation for a unit of data capable of being transmitted at once. A unit of data capable of being transmitted at once can be either one byte (8 bits), one half-word (16 bits), one word (32 bits) or some other definable value. Values registered in the registers are associated with commands from the CPU 100 or commands from an external controller.

Please replace paragraph beginning at **page 4, line 7**, with the following amended paragraph.

The DMA device 104 performs only the function of transferring data stored in the first storage unit to the second storage unit. However, a request can be made so that data having another form different from a form of data stored in the first storage unit is stored in the second storage unit. For example, a request can be made so that data stored in the first storage unit is shifted by the predetermined number of bits, and the shifted data must be stored in the second storage unit. In this case, the DMA device conventionally performs only a control operation for transferring data without carrying out a bit shift operation for data, and the CPU conventionally shifts the transferred data by the predetermined number of bits. Without using the DMA device, the CPU shifts data by the predetermined number of bits and transfers the shifted data to another storage unit. As described above, a transmission rate in the case ~~were~~where the CPU shifts the

data by the specified number of bits, and transfers the shifted data to another storage unit is lower than that in the case where the DMA device transfers the data. The DMA device performs part of the CPU functions so that the CPU load is reduced. However, there is a problem in that the CPU load cannot be reduced since the CPU directly performs an operation of shifting data by the specified number of bits. Thus, a new method for shifting the data by the specified number of bits and transferring the shifted data irrespective of the CPU is required.

Please replace the Table 1 on **page 8** with the following amended Table 1.

TABLE 1

Control register			
Bit number	Bit name	Role	Function
11 to 13	Shift counter	Decision to set number of bits to be shifted	Shift counter: 0 to 7
10	Shift direction	Decision to set shift direction	0: Right shift 1: Left shift
9	Shift enable	Decision to set bit shift operation	0: Non-shift~~Shift~~ operation 1: Shift~~Non-shift~~ operation
8	Endian	Decision to set Endian conversion operation	0: Non-Endian conversion operation 1: Endian conversion operation
7	Destination address direction	Decision to set destination address increment/decrement operation	0: Increment 1: Decrement
6	Source address direction	Decision to set source address increment/decrement operation	0: Increment 1: Decrement
5	Destination address fix	Decision to set destination address fix	0: Increment or decrement 1: Fix
4	Source	Decision to set source	0: Increment or

	address fix	address fix	decrement 1: Fix
2 and 3	Transmission data size	Decision to set unit of transmission bits	00: 8 bits 01: 16 bits 10: 32 bits
1	DMA mode	Decision to set DMA mode	0: S/W 1: H/W
0	DMA activation	Decision to set DMA activation	0: Non-DMA operation 1: DMA operation

Please replace paragraph beginning at **page 20, line 3**, with the following amended paragraph.

FIG. 7 is a view illustrating a procedure for shifting data by means of a protocol layer in accordance with an embodiment of the present invention. FIG. 7 shows a data transmission operation performed by ~~ana~~ radio link control (RLC) layer or a medium access control (MAC) layer. When the RLC layer or the MAC layer desires to attach a header to data, data is shifted by the number of bits corresponding to the header and the shifted data is transmitted. When the bit shift operation is performed by the DMA device rather than the CPU in accordance with an embodiment of the present invention, the processing rate can be enhanced. Table 9 shows the results of a comparison between the data processing rate associated with the CPU and the data processing rate associated with the DMA device.